What is claimed is:

1. An encoder comprising:

a wavelet transform unit operable to convert video data from space domain to wavelet domain;

a half-pixel data generator coupled to receive wavelet domain data from the wavelet transform unit, wherein the half-pixel data generator generates half-pixel data from the wavelet domain data, the half-pixel data approximating wavelet domain data resulting from a wavelet transform of matrices that are offset relative to matrices that the wavelet transform unit transforms; and

a motion estimation unit operable to search the wavelet domain data and the half-pixel data for a block best matching an object block in a frame being coded.

2. The encoder of claim 1, wherein the half-pixel data generator generates the half-pixel data from the wavelet domain data without performing an inverse wavelet transform.

3. The encoder of claim 1, further comprising:

a quantization unit coupled to quantize the wavelet domain data for encoding of intra frames;

a memory; and

a dequantization unit coupled to dequantize data output from the quantization unit and store results in the memory for use in motion estimation for a subsequent frame.

4. The encoder of claim 1, wherein the wavelet transform is a two-dimensional H-transform.

5. The encoder of claim 1, wherein the half-pixel generator generates the half-pixel data for a level of a multi-resolution data structure using high frequency wavelet domain data for the level and an approximation of low frequency wavelet domain data for the level.

6. The encoder of claim 5, wherein the approximation of the low frequency wavelet domain data for the level is determined from low frequency wavelet domain data from a higher level of the multi-resolution data structure.

7. The encoder of claim 5, wherein the higher level of the multi-resolution data structure is a top level of the multi-resolution data structure.

8. The encoder of claim 1, wherein for an offset matrix that is horizontally offset from matrices corresponding to the wavelet domain data, the encoder determines half-pixel data including half pixel coefficients $hxx(r,s)$, $hyx(r,s)$, and $hdx(r,s)$ using the following equations:

$$hxx(r,s) = (h0's - hxs - h0'r - hxr)/2;$$

$$hyx(r,s) = (hys - hds + hyr + hxr)/2; \text{ and}$$

$$hdx(r,s) = (hys - hds - hyr - hdr)/2, \text{ wherein:}$$

r and s are indices identifying arrays that are horizontally offset by one coefficient from the offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r; and

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s.

9. The encoder of claim 1, wherein for an offset matrix that is vertically offset from matrices corresponding to the wavelet domain data, the encoder determines half-pixel data including half pixel coefficients $hxy(r,s)$, $hyy(r,s)$, and $hdy(r,s)$ using the following equations:

$$hxy(r,s) = (hxs - hds + hxr + hdr)/2;$$

$$hyy(r,s) = (h0's - hys - h0'r - hyr)/2; \text{ and}$$

$$hdy(r,s) = (hxs - hds - hxr - hdr)/2, \text{ wherein:}$$

r and s are indices identifying arrays that are vertically offset by one coefficient from the

offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r; and

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s.

10. The encoder of claim 1, wherein for an offset matrix that is diagonally offset from matrices corresponding to the wavelet domain data, the encoder determines half-pixel data including half pixel coefficients $hxy(r,s)$, $hyy(r,s)$, and $hdy(r,s)$ using the following equations:

$$d' = (h0'r + hxr + hyr + hdr)/2;$$

$$g' = (h0's - hxs + hys - hds)/2;$$

$$j' = (h0't + hxt - hyt - hdt)/2;$$

$$m' = (h0'u - hxu - hyu + hdu)/2;$$

$$hxd(r,u) = (m' - j' + g' - d')/4;$$

$$hyd(r,u) = (m' + j' - g' - d')/4; \text{ and}$$

$$hdd(r,u) = (m' - j' - g' + d')/4, \text{ wherein:}$$

r, s, t, and u are indices identifying arrays that are diagonally offset by one coefficient from the offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r;

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s;

hxt, hyt, and hdt are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to t; and

hxu, hyu, and hdu are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to u.

11. An encoding process comprising:

performing a wavelet transformation on a first frame of video data to generate first wavelet domain data;

generating half-pixel data from the first wavelet domain data, wherein the half-pixel data approximates wavelet domain data resulting from a wavelet transform of matrices that are offset relative to matrices used during the wavelet transform;

searching wavelet domain data and the half-pixel data for a block that best matches an object block from second wavelet domain data representing a second frame; and

including in a data stream representing the second frame a motion vector that identifies the block that best matches the object block.

12. The encoding process of claim 11, wherein generating half-pixel data from the first wavelet domain data is performed without performing an inverse wavelet transform.

13. The encoding process of claim 11, wherein:

the first wavelet domain data includes a plurality of levels including a top level and lower levels, each lower level including high frequency wavelet domain data, the top level including high frequency wavelet domain data and low frequency wavelet domain data; and

generating half-pixel data from the first wavelet domain data comprises combining high frequency wavelet domain data from one of the lower levels with an approximation of low frequency for the lower level.

14. The encoding process of claim 13, wherein the approximation of low frequency for the lower level is derived from the low frequency wavelet domain data in the top level.

15. The encoding process of claim 11, wherein generating the half-pixel data comprises determining half pixel coefficients hxx(r,s), hyx(r,s), and hdx(r,s) for an offset matrix that is horizontally offset from matrices corresponding to the wavelet domain data, using the following equations:

$$hxx(r,s) = (h0's - hxs - h0'r - hxr)/2;$$

$$hyx(r,s) = (hys - hds + hyr + hxr)/2; \text{ and}$$

$$hdx(r,s) = (hys - hds - hyr - hdr)/2, \text{ wherein:}$$

r and s are indices identifying arrays that are horizontally offset by one coefficient from the offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r; and

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s.

16. The encoder of claim 11, wherein generating the half-pixel data comprises determining half pixel coefficients hxy(r,s), hyy(r,s), and hdy(r,s) for an offset matrix that is vertically offset from matrices corresponding to the wavelet domain data, using the following equations:

$$hxy(r,s) = (hxs - hds + hxr + hdr)/2;$$

$$hyy(r,s) = (h0's - hys - h0'r - hyr)/2; \text{ and}$$

$$hdy(r,s) = (hxs - hds - hxr - hdr)/2, \text{ wherein:}$$

r and s are indices identifying arrays that are vertically offset by one coefficient from the offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r; and

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s.

17. The encoder of claim 11, wherein generating the half-pixel data comprises determining half pixel coefficients hxy(r,s), hyy(r,s), and hdy(r,s) for an offset matrix that is diagonally offset from matrices corresponding to the wavelet domain data, using the following

equations:

$$d' = (h0'r + hxr + hyr + hdr)/2;$$

$$g' = (h0's - hxs + hys - hds)/2;$$

$$j' = (h0't + hxt - hyt - hdt)/2;$$

$$m' = (h0'u - hxu - hyu + hdu)/2;$$

$$hxd(r,u) = (m' - j' + g' - d')/4;$$

$$hyd(r,u) = (m' + j' - g' - d')/4; \text{ and}$$

$$hdd(r,u) = (m' - j' - g' + d')/4, \text{ wherein:}$$

r, s, t, and u are indices identifying arrays that are diagonally offset by one coefficient from the offset matrix;

hxr, hyr, and hdr are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to r;

hxs, hys, and hds are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to s;

hxt, hyt, and hdt are wavelet domain coefficients and h0'r is an approximate low frequency wavelet domain coefficient corresponding to t; and

hxu, hyu, and hdu are wavelet domain coefficients and h0s is an approximate low frequency wavelet domain coefficient corresponding to u.

18. A video decoding process comprising:

decoding a data structure to extract a difference array and motion vectors associated with a first frame of video, the difference array containing a difference between wavelet domain data;

for each motion vector, identifying whether the motion vector is of a first type or a second type, wherein:

motion vectors of the first type identify respective blocks of wavelet domain data representing a second frame; and

motion vectors of the second type identify respective blocks of data

-25-

corresponding to an approximation of a wavelet transform of a matrix of pixel values representing a portion of the second frame, the matrix being offset relative to matrices used in generating the wavelet domain data representing the second frame;

adding the respective blocks of wavelet domain data identified by motion vectors of the first type to respective portions of the difference array; and

for each motion vector of the second type, generating from the wavelet domain data representing the second frame the block of data identified by the motion vector of the second type, and adding the block thus generated to a portion of the difference array corresponding to the motion vector of the second type.

19. The method of claim 18, further comprising performing an inverse wavelet transform on an array that results from adding the blocks of wavelet domain data and generated data to the difference array.

20. The method of claim 18, wherein the blocks of data identified by the motion vectors of the second type are generated from the wavelet domain data for the second frame without performing an inverse wavelet transform on the wavelet domain data for the second frame.

21. The method of claim 18, wherein:

the wavelet domain data representing the second frame includes a plurality of levels including a top level and lower levels, each lower level including high frequency wavelet domain data, the top level including low frequency wavelet domain data; and

generating the block of data identified by the motion vector of the second type comprises combining high frequency wavelet domain data from one of the lower levels with an approximation of low frequency for the lower level.

22. The method of claim 21, wherein the approximation of low frequency for the lower level is derived from the low frequency wavelet domain data in the top level.

23. A data structure representing video comprising:

first data indicating wavelet domain data for a first frame of the video;

second data indicating a difference array for a second frame of the video;

motion vectors of a first type, wherein the motion vectors of the first type correspond to respective blocks in the difference array and identify blocks of the wavelet domain data for addition to the respective blocks in the difference array when generating wavelet domain data for the second frame; and

motion vectors of a second type, wherein the motion vectors of the second type correspond to respective blocks in the difference array and identify blocks of half-pixel data for addition to the respective blocks in the difference array when generating the wavelet domain data for the second frame, the half-pixel data approximating wavelet transforms of arrays from the first frame that are offset relative to arrays from the first frame that were used in generating the wavelet domain data for the first frame.

24. The data structure of claim 23, wherein the first data comprises:

data indicating a second difference array for the first frame of the video;

motion vectors of the first type that are associated with the first frame, wherein the motion vectors of the first type associated with the first frame correspond to respective blocks in the second difference array and identify blocks of wavelet domain data for a third frame for addition to the respective blocks in the second difference array when generating the wavelet domain data for the first frame; and

motion vectors of the second type that are associated with the first frame, wherein the motion vectors of the second type that are associated with the first frame correspond to respective blocks in the second difference array and identify blocks of half-pixel data for addition to the respective blocks in the second difference array when generating the wavelet domain data for the first frame, the half-pixel data approximating wavelet transforms of arrays from the third frames that are offset relative to arrays from the third frame that were used in generating the wavelet domain data for the third frame.

-27-